

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

**Absolvovanie individuálnej odbornej
praxe**

**Individual Professional Practice in the
Company**

Zadání bakalářské práce

Student:

Martin Bielik

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Absolvování individuální odborné praxe
Individual Professional Practice in the Company

Jazyk vypracování:

čeština

Zásady pro vypracování:

1. Student vykoná individuální praxi ve firmě: Tieto Czech s.r.o.
2. Struktura závěrečné zprávy:
 - a) Popis odborného zaměření firmy, u které student vykonal odbornou praxi a popis pracovního zařazení studenta.
 - b) Seznam úkolů zadaných studentovi v průběhu odborné praxe s vyjádřením jejich časové náročnosti.
 - c) Zvolený postup řešení zadaných úkolů.
 - d) Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe.
 - e) Znalosti či dovednosti scházející studentovi v průběhu odborné praxe.
 - f) Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení.

Seznam doporučené odborné literatury:

Podle pokynů konzultanta, který vede odbornou praxi studenta.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Pavel Moravec, Ph.D.**

Konzultant bakalářské práce: Ing. Roman Šimeček

Datum zadání: 01.09.2015

Datum odevzdání: 29.04.2016

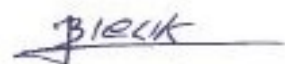

doc. Dr. Ing. Eduard Sojka
vedoucí katedry




prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne. Uviedol som všetky literárne
pramene a publikácie, z ktorých som čerpal.

V Ostrave 21. apríla 2016



Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava.

V Ostravě 25. dubna 2016



Tieto Czech s.r.o.
28. října 3346/91
702 00 Ostrava - Moravská Ostrava
IČO 64608051 DIČ CZ64608051

Rád by som na tomto mieste poďakoval všetkým, ktorí mi s prácou pomohli, pretože bez nich by táto práca nevznikla.

Abstrakt

Moja bakalárska prax bola vykonávaná v firme Tieto czech s.r.o, kde som pôsobil na pozícii software developera. Úlohou tejto praxe bolo spolupracovať na pridelených projektoch a získať čo najviac skúseností popri štúdiu technológie SharePoint.

Na začiatku praxe som dostával jednoduchšie úlohy, ktoré ma mali zoznámiť s technológiou a projektom. Prvé úlohy sa týkali hlavne získavania poznatkov z kódu a jeho menších úprav. Neskôr náročnosť úloh stúpala, od väčších úprav, cez zmeny funkcionality, až k návrhu nového riešenia a jeho zrealizovaním. Na konci prvej časti praxe som sa stretol aj s spísaním funkčnej špecifikácie a pripravením časti prezentácie pre zákazníka.

V druhej časti som sa stretol hlavne s návrhom riešenia problému, zvolením správnej varianty a jeho realizáciou. Výsledkom druhej časti boli samostatné skripty, plniace dané úlohy.

Kľúčové slová: SharePoint, Tieto, User Control, WebPart, Use-Case, chyba, Tieto Teamer, Tieto Share 1.0, Drink

Abstract

My bachelor practice was conducted in Tieto czech s.r.o., where I worked as a software developer. The goal of this practice was to cooperate on assigned projects and to earn experience by studying the SharePoint technology.

At the beginning of the practice I worked on easier tasks that were supposed to familiarize me with the technology and the whole project. The first tasks involved examining the project's source code and making changes in it. Later the difficulty of my tasks increased. I made bigger changes to the code, changed the program's functionality and designed and realized a new solution from scratch. At the end of the first part of my practice I also had to write a functional specification and prepare a part of a presentation for a customer.

In the second part I was mainly analyzing, designing and implementing the right solution for a given problem. The results of the second part were individual scripts that solved the given tasks.

Key Words: SharePoint, Tieto, User Control, WebPart, Use-Case, Bug, Tieto Teamer, Tieto Share 1.0, Drink

Obsah

Zoznam použitých skratiek a symbolov	8
Zoznam obrázkov	9
Zoznam tabuliek	10
Zoznam výpisov zdrojového kódu	11
1 Úvod	12
1.1 Motivácia pre prácu vo firme Tieto	12
2 Popis firmy Tieto s.r.o.	13
2.1 O spoločnosti Tieto	13
2.2 Popis pracovného zamerania	13
3 Využitie technológií	14
3.1 Microsoft SharePoint	14
3.2 Scrum metóda vyvíjania produktu	15
4 Zoznam pridelených úloh počas bakalárskej praxe	16
4.1 Katrin - TFS	16
4.2 Migration - Jira	17
5 Zadané úlohy a postup ich riešenia	18
5.1 Katrin Migration Project	18
5.2 Migrácia Tieto Stránok	25
6 Celkové zhodnotenie praxe	34
6.1 Uplatnenie teoretických a praktických znalostí nadobudnutých v škole	34
6.2 Chýbajúce znalosti a schopnosti	34
6.3 Nadobudnuté znalosti	34
7 Záver	35
Literatúra	36

Zoznam použitých skratiek a symbolov

SP	– SharePoint
WP	– Web Part
JS	– JavaScript
TS	– Tieto Share
TFS	– Team Foundation Server
VS	– Visual Studio 2013

Zoznam obrázkov

1	Logo Tieta	13
2	Logo SharePoint 2013	14
3	Vizuálna ukážka Scrum metódy vyvíjania software	15
4	Ukážka vyhľadávania v How To Buy WP	18
5	Vyplnený požiadavok na kontakt	19
6	Časť ukážky riešenia zmeny grafiky	21
7	Celkový rozpis na priebeh migrácie	25
8	Ukážka štruktúry zložiek	28

Zoznam tabuliek

1	Úlohy pridelené na projekte 'Katrín' a ich časová náročnosť	16
2	Úlohy pridelené na projekte 'Migrácia Tieto Stránok' a ich časová náročnosť . . .	17

Zoznam výpisov zdrojového kódu

1	Získanie korektnej jazykovej verzie	20
2	Ukážka Renderovania VideoPlayer WP	22
3	Príklady Mapovania WP a stránky	23
4	Testovanie Užívateľov oproti AD tieta	26
5	Testovanie Drink attribute oproti AD tieta	27
6	Ukážka kódu na tvorbu zložkovej štruktúry	28
7	Ukážka pridania bufferovania súborov	29
8	Ukážka sťahovania attachmentov k listu	31
9	Ukážka kódu filtrujúcich Administrátorov	32
10	Ukážka kódu riešenia problému s nesprávne uloženými súbormi	32

1 Úvod

Pre moju bakalársku prax som si zvolil firmu Tieto czech s.r.o [1], ktorá ponúkala stáž na pozícii SP developera a neskôr možnosť práce. V tejto spoločnosti som pracoval už predtým, moje doterajšie znalosti sa týkali hlavne zoznámenia s technológiou SP a jeho základných funkcionalít.

V prvej časti opíšem spoločnosť Tieto a popis pracovného zamerania. V druhej stručne popíšem technológiu SP a postup riešenia úloh systémom scrum. V tretej spomeniem úlohy, ktoré mi boli pridelené a projekty, na ktorých som spolupracoval. K vypracovaným úlohám priložím aj spôsob postupu, návrh riešenia a potrebný čas na ich realizáciu. Zahrniem aj ukážky kódu a obrázky jednotlivých funkcionalít.

Na konci zhodnotím nadobudnuté vedomosti, využité znalosti zo štúdia v škole a celkové zhodnotenie môjho pôsobenia vo firme počas bakalárskej praxe.

1.1 Motivácia pre prácu vo firme Tieto

Oproti ostatným spoločnostiam, v ktorých som pôsobil, ponúkala firma Tieto dobrý kariérny rast, možnosť voľby projektu na ktorom chcem pracovať a mnoho ďalších benefitov, medzi ktorými bola napríklad voľná pracovná doba, možnosť práce z domu či firemný laptop.

Začiatkom mojej developerskej kariéry bola Tieto Academy – Code Different. Tu som sa zoznámil s prostredím, kolektívom a pracou v Tiete, ktorá bola pre mňa atraktívnejšia ako v ostatných firmách. Po absolvovaní tohto kurzu mi firma navrhla miesto stáže a možnosť bakalárskej praxe. Po skončení zimného semestra bakalárskej praxe som dostal ponuku na zamestnanecký pomer. Popri bakalárskej praxe v letnom semestri som aj naďalej spolupracoval s firmou Tieto na plný úväzok.

2 Popis firmy Tieto s.r.o.

2.1 O spoločnosti Tieto

Spoločnosť vznikla spojením fínskej firmy Tieto Corporation a švédskej Enator AB pod názvom TietoEnator. Do Českej republiky firma vstúpila v roku 2001 a v roku 2004 otvorila svoje softwarové centrum v Ostrave. V roku 2008 bola spoločnosť premenovaná na Tieto. Mottom firmy je "knowledge, passion, results" (znalosti, vášeň pre vec, výsledky). Firma Tieto aktuálne zamestnáva 14000 zamestnancov na celom svete, z ktorých 2100 pracuje v Českej Republike. Jej zameraním je hlavne poskytovanie IT služieb veľkým a stredne veľkým organizáciám. Aktuálne je najväčším dodávateľom IT služieb pre súkromný a verejný sektor v Škandinávií. Z hľadiska kmeňových zamestnancov je Česká pobočka tretou najväčšou pobočkou Tietu na svete.



Obr. 1: Logo Tietu

2.2 Popis pracovného zamerania

Bol som zaradený do SP tímu, ktorý mal 16 členov a aktuálne pracoval na zákazníckych projektoch. Od začiatku praxe som bol nasadený na projekt Katrin, ktorý bol spravovaný v Team foundation Server. Tento projekt bol v konečnej fáze (dva mesiace do nasadenia), takže som si prešiel úlohami od developmentu, cez opravu chýb, až po nasadenie u zákazníka a spísanie technickej dokumentácie.

Ďalším projektom (interný projekt), ktorý bol spravovaný cez Jira, bola migrácia SP stránok Tietu z Tieto Teamer -> TS 2.0 alias Alfresco. Mojou úlohou na tomto projekte bolo nahradiť stávajúcich členov a uvoľniť ich na projekty pre zákazníkov.

Na začiatku som dostával jednoduchšie úlohy zamerané hlavne na zoznámením s technológiou. Neskôr mi boli pridelené náročnejšie úlohy zamerané na pokročilé vedomosti jazyka C#. Na konci prvého projektu som sa stretol aj s spisovaním dokumentácie pre zákazníka a nasadzovanie riešenia.

Spolupráca s tímom CEM (oddelenie SP) sa mi páči a myslím že do kolektívu som zapadol hneď od začiatku môjho pôsobenia v firme Tieto. Dúfam, že aj naďalej budem prínosom pre môj tím a firmu Tieto.

3 Využíte technológie

3.1 Microsoft SharePoint

Je aplikačná platforma pre web, vyvinutá spoločnosťou Microsoft. Pre programovanie backend kódu som využíval program VS.

Najnižšou verziou SP [2], s ktorou som sa stretol bola verzia SP 2007:

- Oprava mnoho produktov z verzie 2003, predstavenie nových vlastností ako napríklad Business Data Catalog alebo InfoPath Form Service.
- Predstavenie Windows SP Services WSS
- Microsoft Office SP Server 2007 – MOSS

Verzia 2010 už obsahovala rozsiahlejšie zmeny oproti verzií 2007:

- Veľké vylepšenia v Form Management, Workflows, Business Data Services...
- Pridaná funkcionálna SP Foundation
- Microsoft SP Server 2010 – MSS

Verzia, s ktorou som na tejto odbodnej praxi pracoval bola 2013 [3]:

- Táto verzia využíva Office 365
- Pridaná kompletná funkcionálna SP Online
- Vizuálne vylepšenia ako napríklad Imaging Web Service, Discussion Board
- Pridaná Drag and Drop funkcionálna

Aktuálne vychádza najnovšia verzia SP 2016.

Možnosti použitia:

- intranet portál
- správa dokumentov a súborov
- sociálne siete
- webové stránky - portály
- vnútrofremné vyhľadávanie



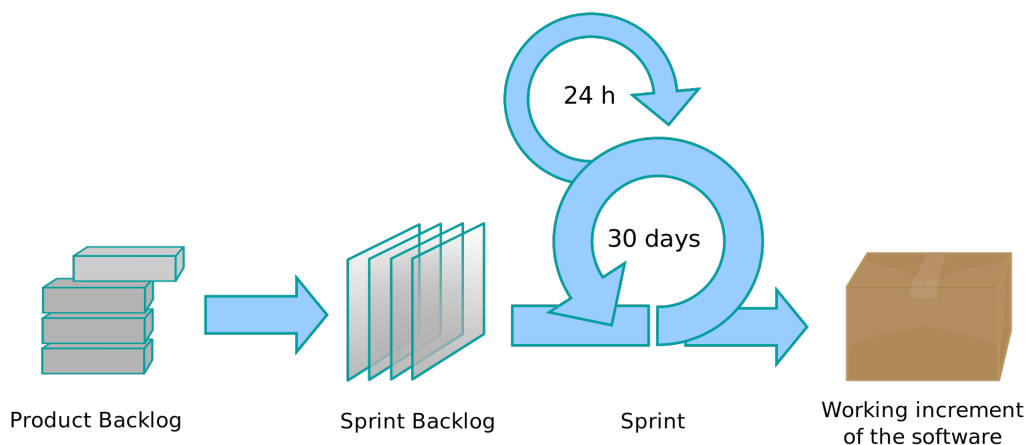
Obr. 2: Logo SharePoint 2013

3.2 Scrum metóda vyvíjania produktu

Scrum [4] metóda zástava tri úlohy.

1. Product owner: Je to človek, na ktorého idú všetky zásadné otázky ohľadom projektu a funkcionality. Product owner pridáva dane požiadavky od zákazníka do backlogu pre programovací tím. Tento človek tvorí business stranu tohto projektu a nie je prítomný na denných mítingoch. Po správnosti by Product Owner nemal komunikovať s programovacím tímom.
2. Development team: Do tejto role som bol zaradený ja. Mojou úlohou bolo byť prítomný na každodenných mítingoch a riešiť problémy v jednotlivých sprintoch. Pri prípadných komplikáciách, som buď kontaktoval mojich kolegov ak išlo o programátorský problém, alebo Scrum Mastera ak išlo o organizačnú vec či návrh na požiadavku zákazníka. Len vo výnimočných prípadoch ma Scrum Master odkázal na Product Ownera, s ktorým som prebral problém, navrhol riešenia a on zvolil jednu z nich.
3. Scrum master: V tejto roli som na mojich dvoch projektoch mal vždy jedného Scrum Mastera, ktorý bol zodpovedný za vedenie každodenných mítingov a riešenie problémov. Tak tiež jeho úlohou bolo mať prehľad ohľadne pokroku na projekte a v konečných fázach projektu aj pomáhať spisovať prezentáciu popisujúcu zmeny.

V Tiete sa väčšina projektov riešia SCRUM metódou vývojom produktu. Na týchto mítingoch sa preberá, čo ktorý pracovník robil predchádzajúci deň a čo bude robiť dnes, prípadne sa diskutovali a riešili vzniknuté problémy v riešení. Tieto mítingy prebiehali samozrejme v anglickom jazyku.



Obr. 3: Vizualna ukážka Scrum metódy vyvíjania software

4 Zoznam pridelených úloh počas bakalárskej praxe

Táto sekcia obsahuje úlohy, ktoré mi boli pridelené počas bakalárskej praxe. Následné tabulky vždy obsahujú číslo úlohy, jej názov, typ a časový odhad jej zrealizovania.

4.1 Katrin - TFS

Projekt Katrin bol riadený v software TFS [6], kde boli každodenne pridelené nové a odoberané hotové úlohy. Všetky zmeny v kóde sa potvrdzovali od všetkých na server a pri vzniknutých nejasnostiach sa spojovali verzie automaticky alebo jednotlivo ručne.

Number	Name	Type	Estimation [h]
1706	Create application page for contact form	Task	18
1707	Create opening of Contact page in lightbox	Task	12
2254	Contact request popup Controls - left and right	Task	10
2565	Migrate existing resources (translations) from old solution and merge them with new	Task	24
2481	How to buy - Contact Finder WP	Task	12
3382	Contact request popup - finalize perfect solution case	Task	6
3386	About Katrin, Contact - List WP	Task	12
3387	About Katrin, Contact - Web, page, layout, WP	Task	12
3578	PowerShell script for insert data into How to buy WP	Task	4
3606	Language picker and change language functionality	Task	20
3812	Main Slider Panel changeto WowSlider solution	Task	8
4063	Create general web template and use it for all missing webs	Task	12
4141	Contact me page : Anti spam check //Azbuka error	Bug	2
4210	Contact dialog two issues, description inside	Bug	2
4214	SiteMap	Task	12
4269	Send mail	Bug	1
4530	Contact me page - page layout displays resource text	Bug	1
4671	Tooltip text when mouseover 'about Katrin'	Bug	1
4805	Anonymous access does not work for Change language feature	Bug	2
4871	Create video WP	Task	6
4922	Create Script for migration static sites	Task	84

Tabuľka 1: Úlohy pridelené na projekte 'Katrin' a ich časová náročnosť

4.2 Migration - Jira

Migrácia Tieto stránok bola riadená v software Jira [5]. Tu sa kód neposielal na server, každý pracovník vlastnil svoj separátny skript vykonávajúci danú funkcionality. Na servery Jira potom boli zobrazené jednotlivé sprinty, fáze a úlohy.

Number	Name	Type	Estimation [h]
77	Stage I - Migrate ZIPs to Somewhere 1.1.2010 - 31.12.2013	Task	84
78	Stage I - Migrate ZIPs to Somewhere xx.xx.xxxx - 31.12.2009	Task	84
293	Get TS 1.0 site list which only has internal Tieto users (No external users)	Task	36
302	Get user Drink attribute status for Phase II (d)	Task	6
316	Pre-requist for Teamer Phase II (e) sites	Task	6
319	Get full list of group for granting access for Phase II (e)	Task	6
323	Finalize ZIP archive script	Task	84
342	Get user Drink attribute for Phase III (a) all version sites	Task	6
344	Get pre-requisites for Phase III (a) all version	Task	6
357	Create script for removing Teamer access for those zipped sites & already migrated sites	Task	84
363	Revert removed user access for passive sites	Task	84

Tabuľka 2: Úlohy pridelené na projekte 'Migracia Tieto Stránok' a ich časová náročnosť

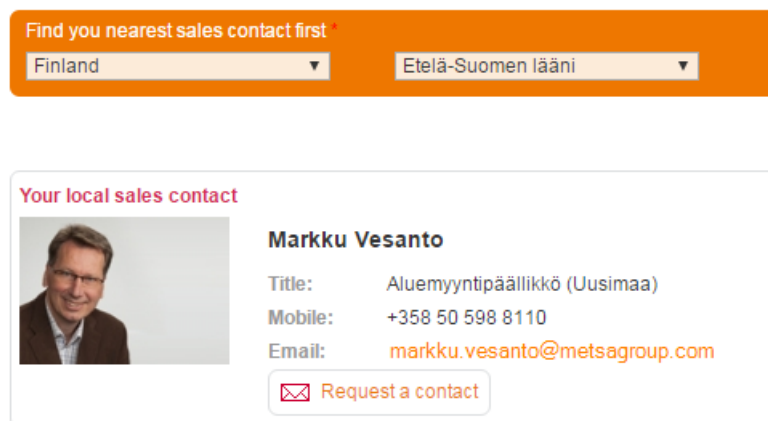
5 Zadané úlohy a postup ich riešenia

Na začiatku môjho pôsobenia na tomto projekte som dostal úlohu, ktorá ma mala zoznámiť s celým projektom a uviesť do tímu. Neskôr sa náročnosť mojich úloh stupňovala. Na konci projektu som sa stal plnohodnotným členom tímu.

5.1 Katrin Migration Project

5.1.1 Ako nakupovať - WP na vyhľadávanie kontaktov

Cieľom tejto úlohy bola migrácia WP na vyhľadávanie kontaktov pre nákup. Po vybraní z ponuky obchodnej oblasti a okrsku, mali byť užívateľovi zobrazené kontakty podľa kritérií. Základom bolo vytvorenie listov a migrácia dát z predošlého produkčného serveru. List sa mal nachádzať na scope "Site". Na migráciu dát som exportoval staré dáta v podobe csv súboru a importoval ich cez mnou vytvorený powershell skript do nového riešenia. Pri migrácii vznikol problém s dátami typu "Lookup", kde riešením bolo správne poradie importovania listov alias 'nemôžeme sa odkazovať na niečo čo neexistuje'. Následne som s príslušnými dátami vytvoril WP a pridal jej User Control ContactList. Ten som potom musel pridať do web.configu v safe controls, pretože bez toho mi tento control nedovolilo spustiť na stránke. ContactList bol hlavným jadrom tejto úlohy. Obstarával načítanie dát do dropdownlistov, zmenu výberu a zobrazenie kontaktov. Pre zobrazenie jednotlivých kontaktov sa používal UserControl. User Control ContactItemService.cs mal v sebe základný kód, ktorý ostatné druhy kontaktov dedil, prípadne prepisovali kľúčové metódy vykonávajúce napríklad rendering na stránku. Druhy kontaktov sa rozdeľovali do kontaktov s obrázkami a kontaktov bez obrázkov. Ďalej podľa špecifického zamerania na kanceláriu, osobu či službu. Nakoniec každý kontakt obsahoval možnosť vyžiadania kontaktu, ktorý následne otvoril okno s informáciami o kontakte danej osoby. Problémy pri tejto úlohe vznikali pri riešení compatibility s verziami SP a zdrojových kódov zo starého riešenia. Každý z týchto problémov som samostatne riešil s mojim mentorom.



Obr. 4: Ukážka vyhľadávania v How To Buy WP

5.1.2 Požiadavok na vybranie kontaktu v otváracom okne

Táto úloha sa zameriavala na riešenie vyžiadania kontaktu z rôznych častí stránky.

Zadané boli 4 Use-Case:

- how to buy
- perfect solution
- my collection
- other

Prvý Use-Case sa zameriaval na prípad how to buy stránky. Po rozkliknutí možnosti na popup (tlačítko Request Contact) sa zavolať SalesLeadFormPage s preddefinovaným Sales Managerom. SalesLeadFormPage musel dediť z UnsecuredPageLayout, aby bol dostupný aj anonymným užívateľom. Po vyplnení a potvrdení užívateľ odoslal email so správou pre daného Sales Managera. Ak užívateľ odškrtnol vyžiadanie kópie, získal na svoju emailovú adresu kópiu svojej správy. Ostatné Use-Casei sa líšili hlavne v pridanej prílohe v emailu, primárne však obsahovali linky na vybrané výrobky.

Obr. 5: Vyplnený požiadavok na kontakt

5.1.3 Preklady jazykových verzií

Na stránke bolo požadovaných 7 jazykových mutácií. S využitím znalosti z AT.NET som použil .resx súbory k uskladneniu jednotlivých prekladov. Samotnú voľbu jazyka som následne ukladal do cookies a pri každej požiadavke kontroloval hodnotu cez http handler (.ashx). Tým bolo zaručené, že zvolený jazyk bude vždy uložený v cookies a môže sa podľa neho prekladať. Pri prvom vstupe na stránku overil systém užívateľovi nastavený jazyk v prehliadači. Ak tento jazyk nebol žiaden z povolených, spustil užívateľovi anglickú verziu. Vznikol tu problém s občasnými anomáliami, kedy sa do cookies neuložilo nič. Tento problém sa vyriešil použitím regexu v handlery, ktorý obstarával korektnosť url.

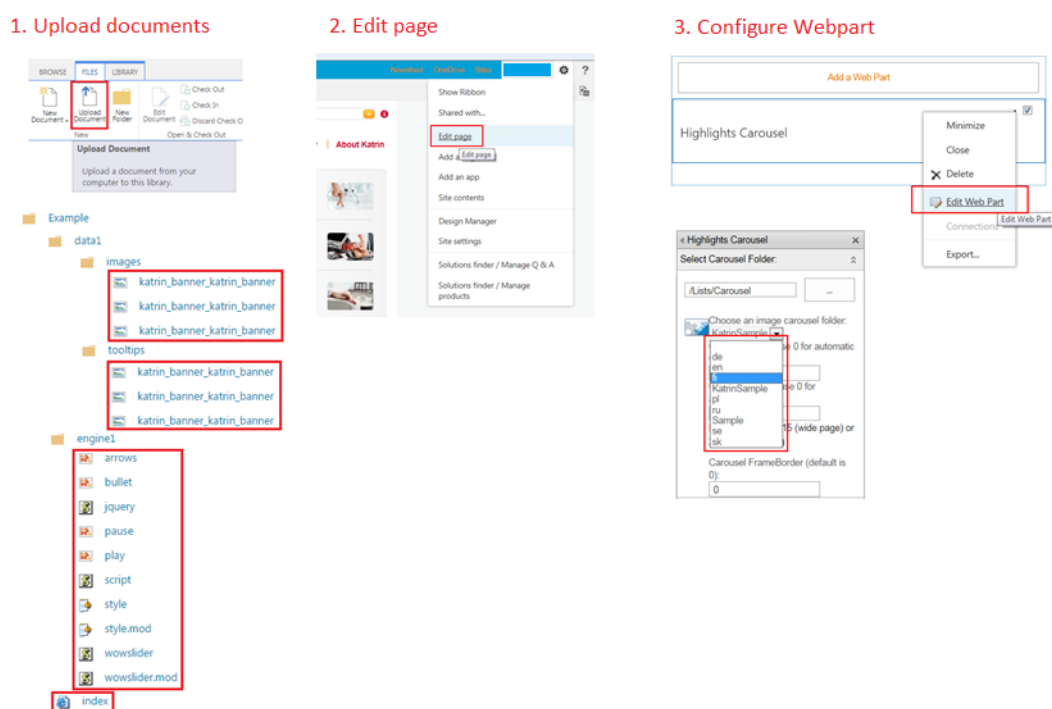
```
private static string GetValidLanguage(string browserLanguagesInput = "")
{
    var result = _DefaultLanguage;
    if (browserLanguagesInput == "")
    {
        var browserLanguages = HttpContext.Current.Request.UserLanguages;
        if (browserLanguages.Any())
            browserLanguagesInput = browserLanguages[0];
    }
    // get collection of languages, format fi:fi-FI:fi, en:en-US:en...
    string[] languages = GetListItemCollection();
    foreach (string item in languages)
    {
        if (item.Split(':')[0].Equals(browserLanguagesInput.ToLower().Substring
            (0, 2)))
            result = item.Split(':')[1];
    }
    return result;
}
```

Výpis 1: Získanie korektnej jazykovej verzie

5.1.4 WP na hlavnej stránke zobrazujúca hlavný baner

Zákazník sa rozhodol pre iný návrh hlavného slideru na stránke. Mojou úlohou bolo stiahnuť si program WowSlider s plnou licenciou kúpenou zákazníkom, nahrať do programu súbory s obrázkami a textami od zákazníka a vygenerovať JS knižnicu WowSlideru. Následne ho implementovať na hlavnú stránku Katrin a vytvoriť riešenie zmeny, napr. vytvorenie obrázku na stránke alebo zmenu celého vygenerovaného JS s WowSliderom (+ dopodrobna popísať v dokumentácii postup

zmeny grafiky). Výsledkom tejto úlohy teda bola zmena doterajšieho riešenia JS, na zákaznikom požadovaný nový WowSlider, ktorý sa nachádzal v iframe.



Obr. 6: Časť ukážky riešenia zmeny grafiky

5.1.5 Vytvorenie SiteMap WP

Tato WP mala byť migrovaná zo starého riešenia do nového bez väčších zmien, kde však vznikali problémy s verziami a následne s mapovaním. Keďže sa jedná o SP predvolenú WP a nie custom WP tak vyriešenie tohto problému netrvalo dlho. Použila sa novšia verzia s upravenou konfiguráciou premenných.

5.1.6 Vytvorenie stránky 'Ako nakupovať' a 'O katrin - kontakt'

Úlohou bolo vytvoriť automatizované vytvorenia všetkých jazykových verzii tejto stránky a správne zmapovať obsah. Vznikol tu problém pri používaní telerik WP, ktorá bola platená a zákazník ju chcel nahradiť neplatenou variantom. Keďže v normálnom textovom poli bolo zakázané používať JS (popup na vyžiadanie kontaktu), tak po konzultácii s mentorom som navrhol použiť skript editor WP, ktorý mohol zahrňovať JS v texte. Nevýhodou tohto riešenia bola nutnosť ručnej konfigurácie WP na každej jazykovej variácii. Po pridaní záznamov do skriptu a upravenie obsahu bolo automatické vytváranie stránok funkčné. Ďalej bolo nutné pridať návod na ručnú konfiguráciu WP na stránkach do ConfigurationSteps súboru.

5.1.7 Vytvorenie základnej predlohy na statické stránky Katrin

V konečnej fáze bola požadovaná site template, ktorá by sa použila pre 90% stránok. Avšak musela byť kompatibilná so všetkými statickými stránkami, ktoré sa mali na ňu namapovať. Template teda v poslednej verzii obsahovala veľa WP zón a vznikol chaos pri mapovaní starých stránok na nové. Po dohodnutí so spolupracovníkmi na pravidlách mapovania bola väčšina stránok úspešne zmigrovaná, avšak stále vznikali problémy s WP pre zobrazenie odkazov na podrobnejšie stránky. Výsledkom bolo verzovanie tejto WP a následne zakomponovanie mapovania do general template.

5.1.8 Vytvorenie a implementácia Video WP

Na rôznych stránkach sa vyskytovali zakomponované videa (Flash Player). Po preskúmaní starého riešenia som našiel licencovanú verziu video prehrávača, ktorú som využil aj pri novom riešení. Vložil som daný prehrávač do knižnice a následne som vytvoril WP, ktorá bude obsahovať property na nastavenie videa a použitie tohto Flash Playeru na stránke. Výsledkom bola potom jednoduchá WP, ktorá zhrnula na stránku kód načítavajúceho property z WP.

```
[Category("Data"), Personalizable(PersonalizationScope.Shared), WebBrowsable(
    true)]
public string _VideoUrl { get; set; }

protected override void CreateChildControls(){
    try{
        base.CreateChildControls();
        VideoPlayerControl myControl = (VideoPlayerControl)Page.LoadControl("~/
            _controltemplates/15/VideoPlayer/VideoPlayerControl.ascx");
        myControl._Title = TitleOnVideo;
        myControl._VideoPreviewImageUrl = VideoPreviewImageUrl;
        myControl._VideoUrl = VideoUrl;
        this.Controls.Add(myControl);
    }
    catch (HttpException ex){
        this.Controls.Add(new LiteralControl("<br />An unexpected error
            occurred loading Web Part. " + ex.Message));
    }
}
```

Výpis 2: Ukážka Renderovania VideoPlayer WP

5.1.9 Vytvorenie skriptu pre migráciu statických stránok a automatizáciu migrácie stránok

Tato úloha bola základom na migrovanie všetkých site v starom SP, kde sa musel vytvoriť program na stiahnutie obsahu a jeho zmapovanie na nové šablony z dôvodu požiadavky zákazníka na responzivitu webu. Vtom čase som bol jediný voľný pracovník na projekte, ktorý poznal nové riešenie. S externým spolupracovníkom, ktorý mal znalosti starého riešenia sme vytvárali skript, ktorý sťahoval obsah, mapoval ho na nové riešenie a nahral stránky do nového riešenia.

Mojou úlohou na tejto spolupráci bolo mapovať staré premenné s novými, názvy WP a riešiť vzniknuté problémy spojené s mapovaním. Príklady týchto problémov boli napr. viacnásobne šablony pre WP, ktoré sa u nás riešili jednou šablou. Na tento špecifický problém sme mali hodinový míting, na ktorom sa špecialistom opísal problém a oni navrhli riešenia.

Výsledkom bolo vytvoriť jednotlivé WP ku každej šabli a pridať pravidlo pri mapovaní, čo sa týka verzie šablony.

```
// Webpart mapping example
WebpartMapping AboutKatrinContactDisplay = new WebpartMapping();
AboutKatrinContactDisplay.OldName = "About Katrin - Contacts";
AboutKatrinContactDisplay.NewName = "AboutKatrinContactDisplay";
AboutKatrinContactDisplay.Zone = "DownloadListing";
AboutKatrinContactDisplay.Index = 1;

// Page mapping example
PageTemplateMapping content2 = new PageTemplateMapping();
content2.OldLayout = "katrincontent2.aspx";
content2.NewLayout = "ContentPageTemplate2.aspx";
content2.PropsMapping.Add(new Tuple<string, string>("ows_TitleOnPage", "
    TitleOnPage"));
content2.PropsMapping.Add(new Tuple<string, string>("ows_MediaContent", "
    PageImage"));
content2.WebParts.Add(AboutKatrinTelerik);
content2.WebParts.Add(AboutKatrinContactDisplay);
content2.WebParts.Add(SummaryLinkWebPart_SectionPageTemplate_Right);
content2.ContentTypeID = "0x0100261BC0D988BC48FCB990176A42A0385E";
pageTemplateMappings.Add(content2.OldLayout.ToLower(), content2);
```

Výpis 3: Príklady Mapovania WP a stránky

5.1.10 Oprava chýb

Súčasťou mojej práce bola aj korekcia vzniknutých chýb. Prvú požiadavku na chybu som dostal od testera na problém pri načítaní predvolenej biznis zóny v WP - how to buy. Problém som vyriešil pridaním overovacej funkcie do onLoad metódy v WP. Ta automaticky vybrala zvolený prvok pri prvom vstupe na stránku. Prvý vstup sa zistil z cache pamäte. Ďalšia chyba sa týkala nastavenia SMTP serveru na posielanie e-mailov. SMTP server mohol byť získaný, buď cestou SP z jeho konfigurácie, alebo cestou asp.netu z web.configu. Ja som zvolil cestu asp.net, ktorá sa ukázala ako chybná kvôli inému nastaveniu SMTP serveru u zákazníka. Po prepracovaní na verziu kódu SP sa smtp server správne načítal a posielanie e-mailov fungovalo korektne.

5.1.11 Dokumentácia

Moja posledná úloha na tomto projekte bola príprava funkčnej špecifikácie prvkov, na ktorých som pracoval. Obsahom tejto dokumentácie malo byť funkčné použitie pre zákazníka a práca s modulmi. Zákazník požadoval napríklad pri WowSlideru návod na pridávanie vlastných obrázkov a textov, kde návod musel obsahovať umiestnenie súborov a nakonfigurovanie WP. Po tom čo si užívateľ vygeneroval dáta z externej aplikácie, musel následne vložiť dané dáta do špecifického listu. Potom bude zákazníkovi sprístupnená možnosť nastavenia WP. Po jej potvrdení budú zmeny viditeľné na WP.

Ukážka dokumentácie Katrin - Ako nakupovať

4.10.10 How to buy page

How to buy page is a section (site) in the web site. The main page of How to buy section includes Contact finder element. The user is directed to How to buy page from other parts of the web site with link elements.

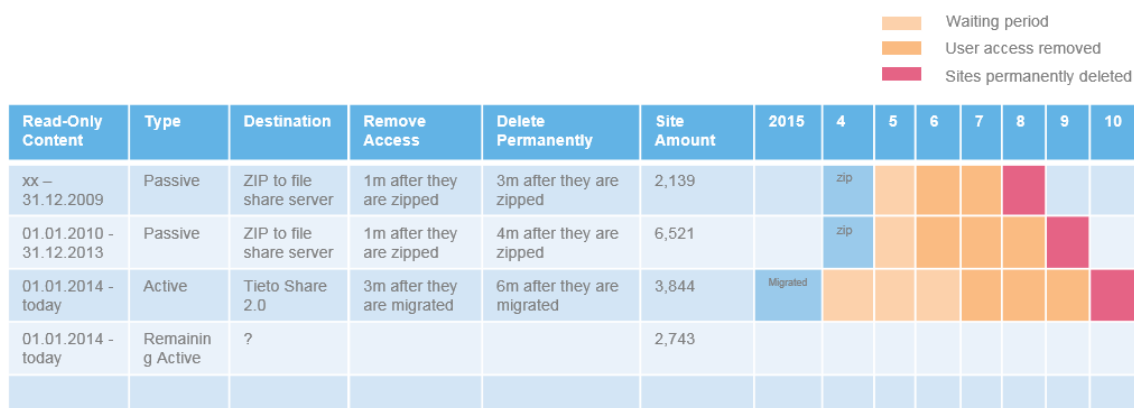
5.2.2 Sales lead form

Lead form is the primary web contact tool for sales. The same form is used for the following use cases:

- *Use case 1: User comes from How to buy page – Information in contact finder is prefilled with the selection made in How to buy page*
- *Use case 2: User comes from Perfect solution page – Perfect solution page's title and products in Perfect solution are sent in the Email message to selected sales contact and to user.*
- *Use case 3: User comes from My collection – List of products in My collection are sent in the Email message to selected sales contact and to user.*
- *Use case 4: User comes somewhere else – The form is in default state and no information is carried over in the background.*

5.2 Migrácia Tieto Stránok

Cieľom tohoto interného projektu bolo migrovať stránky z Tieto Teamer a TS 1.0 do TS 2.0 (Alfresco) systému. Ja osobne som bol nasadený na pasívne stránky, kde som mal pripraviť acl k stránke a jej obsah v zip formáte. Popri tom som pomáhal tímu so skriptom na zisťovanie užívateľov, ktorí sú aktívni na určitých fázach stránok, prípadne ktorým sa objedná tzn. Drink attribute alebo na získanie čisto interných stránok zo starého produkčného serveru Tieto.Share 1.0. Projekt prebiehal po týždňových fázach, kde v jednotlivých fázach migrovali stránky, rozposielali emaily, mrazili stránky a riešili problémy vzniknuté pri migrácií.



Obr. 7: Celkový rozpis na priebeh migrácie

5.2.1 Skript na získanie interných stránok z Tieto.Share 1.0

Tento skript mal slúžiť na vyselektovanie stránok, ktoré obsahovali len interných užívateľov (nie externých). Zoznam všetkých stránok na servere Tieto.Share 1.0 som získal pomocou powershell skriptu, ktorý iteroval všetky podweby daného webu. Keď narazil na možnosť podwebu, vstúpil doňho, zapísal záznam a znovu sa dotazal na podweb, ak žiadny nenašiel, vrátil sa o úroveň vyššie. Potom čo mi tento skript dodal výsledný súbor s url adresami všetkých Tieto.Share 1.0 stránok, som pripravil skript, ktorý načítaval všetkých užívateľov na stránke. Tento skript následne otestoval každého užívateľa v Active Directory Tieto(AP/EU doména). Ak sa tam tento užívateľ nachádzal, znamenalo to, že užívateľ je interný, v ostatných prípadoch bol externý. Pokiaľ mala stránka len interných alebo žiadnych užívateľov bola zaradená do zoznamu interných stránok, ak mala stránka aspoň jedného externého užívateľa, bola zaradená do zoznamu externých stránok.

```
string domain = login.Split('\\')[0];
string path = "LDAP://" + domain;

//init a directory entry
DirectoryEntry dEntry = new DirectoryEntry(path);

//init a directory searcher
DirectorySearcher dSearcher = new DirectorySearcher(dEntry);
dSearcher.Filter = String.Format("&(sAMAccountName={0})", login.Split(
    '\\')[1]);

//perform search on active directory
var sResults = dSearcher.FindAll();
if (sResults.Count > 0){
    var a = sResults[0].Properties["displayName"];
    return a[0].ToString();
}
return "";
```

Výpis 4: Testovanie Užívateľov oproti AD tieta

5.2.2 Skript na zistenie stavu užívateľov

Tento skript bol potrebný z dôvodu nedostatku licencií. Mal zhodnotiť, či je užívateľ aktívny a objedná sa pre neho drink attribute. Na začiatok som si pripravil skript na iterovanie stránok, potom na interovanie všetkých užívateľov, skupiny a užívateľoch v skupinách. Tým som dostal užívateľov použitých na stránkach. Vznikol tu problém s domain skupinami (so skupinami domény) SP. Tieto skupiny sa totiž správali ako užívatelia v SP, ale ako skupiny v AD.

Riešením tohto problému bolo vyhľadať všetky grupy v AD a porovnať ich s danou domain skupinou. Na toto som použil powershell skript, ktorý kontroloval všetky krajiny a filtroval systémové skupiny. Takisto som použil aj powershell skript na získanie všetkých užívateľov Tieta, ktorí sú aktívni. Oproti týmto zoznamom som potom jednotlivo porovnával užívateľov získaných zo SP stránok. Určití phase.užívatelia, ktorých mi skript zaradil do súboru ActiveUsers.csv alebo DLgroups.csv, získali email o migrácii ich stránky.

Tato stránka bola následne po týždni zmrazená a premiestnená do Alfresca. Ďalej som porovnával, či títo aktívni užívatelia mali kúpený tzn. drink attribute. Ak nemali, bola odoslaná požiadavka na jeho zakúpenie, pretože len užívatelia s drink attribute môžu vstúpiť do alfresco systému, avšak grupy drink atribút neobsahovali.

```
string domain = login.Split('\\')[0];
string path = "LDAP://" + domain;

// init a directory entry
DirectoryEntry dEntry = new DirectoryEntry(path);

// init a directory searcher
DirectorySearcher dSearcher = new DirectorySearcher(dEntry);
dSearcher.Filter = String.Format("&(objectCategory=person)(objectClass=user)(sAMAccountName={0})(employeeID=*)(!(userAccountControl:1.2.840.113556.1.4.803:=2)))", login.Split('\\')[1]);

// perform search on active directory
var sResults = dSearcher.FindAll();
if (sResults.Count > 0){
    return true;
}
else{
    return false;
}
```

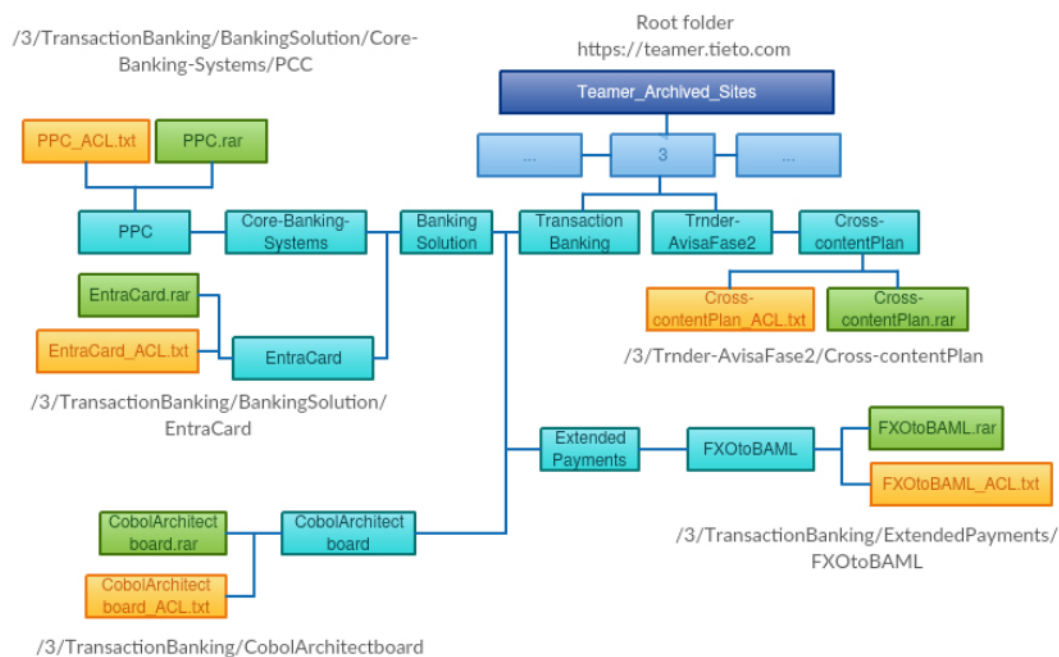
Výpis 5: Testovanie Drink attribute oproti AD tieta

5.2.3 Skript na stiahnutie obsahu a zipovanie na archiváciu

Cieľom tohto skriptu bolo navrhnúť spôsob zipovania pasívnych stránok (okolo 8000) v Tieto Te-amer. Tieto stránky boli vypísané skriptom môjho spolupracovníka, ktorý vyhodnocoval stránky, na ktorých bola posledná zmena zaznamenaná užívateľom pred viac ako 3 rokmi. Tieto zipy sa mali uložiť do úložiska v sieti.

Návrh usporiadania jednotlivých site

Mojim návrhom bolo použiť url adresu stránky a podľa znaku '/' vytvárať zložkovú štruktúru . Posledná zložka s menom webu potom obsahovala zip s obsahom a acl file.



Obr. 8: Ukážka štruktúry zložiek

```
public static void CreateDirectoriesInLocal(string path){
    string[] pathParts = path.Split('/');
    for (int i = 0; i < pathParts.Length; i++)
    {
        if (i > 0)
            pathParts[i] = Path.Combine(pathParts[i - 1], pathParts[i]);
        if (pathParts[0] == "" || pathParts[0] == null)
        {
            pathParts[0] = ConfigurationSettings.AppSettings["PrerequisitesPath"];
            continue;
        }
        // add folders to local
        if (!Directory.Exists(pathParts[i]))
            Directory.CreateDirectory(pathParts[i]);
    }
}
```

Výpis 6: Ukážka kódu na tvorbu zložkovej štruktúry

Návrhy na spôsob ukládania

Prvý návrh bol ukládať stránky na jednom zo 4 frontendov, kde bola 20gb lokálna pamäť k dispozícii a následne ich premiestniť cez sieť na úložisko, kde bolo dostatok miesta na uloženie všetkých stránok. Nevýhodou tohto postupu by bola neustála fixácia na beh programu. Výhodou bola istá korekcie dát kvôli malému počtu prenesených stránok v jednotlivých cykloch z dôvodu nedostatku miesta na lokálnom úložisku. Vznikal tu však problém so stránkami o veľkosti väčšej ako 20gb.

Môj návrh na takéto stránky bol prerušiť program počas zipovania, preniesť súbor, zmazať ho na predchádzajúcom úložisku a následne obnoviť beh programu. Tento nápad sa mi ale nepodarilo zrealizovať, pretože vytváranie zip súboru o veľkosti 15gb nedovolí zmazať predchádzajúci zip súbor a vytvárať ďalší.

```
if (!ZIP.EntryFileNames.Contains(_pathInZip))
{
    Z.ZipEntry entry = ZIP.AddEntry(_pathInZip,
zipStreamFiles.First(x => x.Length.Equals(fileFromSP.Length)));

    // counter for buffering
    bufferForZipping += (int)fileFromSP.Length;

    // List of stored entries, about adding entries to zip
    zipEntries.Add(entry);
}
// 500mb buffering
if (bufferForZipping > 500000000)
{
    ZIP.Save();

    // refresh memory
    zipEntries = new List<ZipEntry>();
    zipStreamFiles = new List<Stream>();
    bufferForZipping = 0;
}
```

Výpis 7: Ukážka pridania bufferovania súborov

Druhý návrh bol ukládať Tieto zipy priamo na úložisko cez sieť, teda vytvárať zipy priamo po sieti. Boli tu obavy z občasných výpadkov siete alebo z rýchlosti komprimovania dát cez sieť. Preto som mal odskúšať aj tento návrh a dodať investigáciu rýchlosti tohto spôsobu migrovania. Občasné výpadky som riešil v skripte handlovaním počas výpadku, takže ak server nebol na chvíľu dostupný, môj program fungoval ďalej a zaznamenal do logu neúspech z migrovania stránky. Na ďalšom spustení tento skript už iba verifikoval súbory. Ak narazil na zmenu (výpadok siete) zmazal zip danej sity a stiahol ju nanovo. Rýchlosť v začiatkových štádiách exponenciálne narastala s veľkosťou stránky alebo počtom položiek na stránke. Ale stále pri aktuálnej rýchlosti zipovania (10GB/50sites-6hours), bol toto správny postup. Po úprave skriptu a pridanie bufferovania súborov sa zipovací proces zrýchlil 20 násobne.

Po voľbe návrhu spôsobu ukladania (priamo na dané úložisko na sieti), vznikali ďalšie problémy v zipovaní. Bolo požadované vytvorenie zložkovej štruktúry rovnakej ako na stránke. Pri niektorých špeciálnych znakoch v názvoch document library alebo folderov, program padol. Zaviedla sa teda `replaceSpecialCharacters` funkcia, ktorá mala Tieto názvy prepísať. Ďalšou požiadavkou pre zipovanie obsahu bolo migrovanie špeciálnych libraries(knižníc) na ukladanie wiki stránky. Tieto stránky boli písané ako aspx kód. Po prediskutovaní tohto problému som sa rozhodol migrovať aspx kód a premennú uchovávajúcu obsah stránky item - `ows_WikiField`.

Ďalšou požiadavkou bolo migrovanie blogov. Po investigácii tejto požiadavky sme zistili, že blogy sú jednotlivé osobitné podweby, takže tento problém bol vyriešený automaticky na začiatku pri získavaní všetkých pasívnych stránok.

Po úspešných testoch na document libraries a wiki stránok prišli na rad listy. Tie mali byť uchovávané v csv súbore. Prázdne listy sa migrovať nemali. Vznikol tu problém s dátami v týchto listoch. Keďže každý list mal vlastné zobrazenia postavené na základe query, ktoré mohlo uchovávať iné dáta aké boli pôvodne v databáze. Problém s dátami z databázy boli mená jednotlivých stĺpcov, na rozdiel od view nejakého listu kde boli mená stĺpcov napr.: Name, Title, Assigned to, Completed(%). V databáze Tieto stĺpce boli pod názvami `nvarch3`, `nvarch31`, `int4`. Z tohto dôvodu som navrhol stiahnuť k danému listu všetky jeho views a prípadne stiahnuť záznam o všetkých dátach z db. Týmto sa malo predísť strate dát a nezrozumiteľnosti v ich vyhľadávaní. Po vyriešení dát z listov vznikol problém s chýbajúcimi prílohami.

Navrhol som, aby všetky attachmenty k listu budú ukladané do zložky s rovnakým názvom, aký má list. Bug report k zipovaniu som dostal na názvy stĺpcov listov. Zatiaľ čo som ja používal predvolenú verziu mien SP, zodpovedný manager pri testovaní narazil na odlišnosti so stĺpcovými menami užívateľov. Po hĺbkovom preskúmaní tohoto problému som zistil, že k menu užívateľa sa dá dostať cez metódu `GetFieldByInternalName()`, kde sa ako argument použil SP predvolený názov, ktorý som mal k dispozícii.

```

CreateDirectoriesInLocal(web.ServerRelativeUrl + "/" + spList.Title);
// create directory with name of List
foreach (string attachment in row.Attachments)
{
    SPFile file = web.GetFile(row.Attachments.UrlPrefix + attachment);

    byte[] binFile = file.OpenBinary();
    System.IO.FileStream fs = System.IO.File.Create(ConfigurationSettings.
        AppSettings["PrerequisitesPath"]
        + web.ServerRelativeUrl + "/" + spList.Title + "/" + attachment);

    fs.Write(binFile, 0, binFile.Length);
    fs.Close();
}

```

Výpis 8: Ukážka stahovania attachmentov k listu

Pre jednoduchšiu prehľadnosť a orientovanie v stránkach bola požiadavka na vytvorenie SiteMap.csv. Dokument Excel obsahujúci stĺpec s URL stránkou, názvom stránky a jej odkaz na sieťové úložisko, po ktorom sa otvorí pri rozkliknutí zložka, kde sa nachádzal zip a acl súbor. Data na odkaz som ukladal do csv s prefixom a suffixom v podobe:

```
=HYPERLINK("\SERVER\FOLDERFORSTORINGDATA\FOLDERSTRUCTURE\FOLDERWITHWEBNAME\")
```

5.2.4 Skript na prerequizity

Tento skript slúžil na prechádzanie stránok a vytváranie textového dokumentu, ktorý obsahoval všetky dáta ohľadom užívateľov použitých na stránkach. Ďalej mal obsahovať záznamy o miestach, kde bola prelomená dedičnosť užívateľov a zoznam užívateľov, ktorý mali mať prístup k týmto špeciálnym miestam na stránke.

Problémom na tejto úlohe bol v prístupe získavania užívateľov a ich role. Pretože niektorí užívatelia boli pridaní na stránku bez role a pritom boli v skupine adminov, takže mali plné práva. Riešením tohto problému bolo zrušenie iterovania roli užívateľa a použitie miesto toho dotaz priamo na web.

```

// Only users with administrator rights
if (web.DoesUserHavePermissions(oUser.LoginName, SPBasePermissions.ManageWeb))
{
    File.AppendAllText(pathToPrerequisites, oUser.Name + ", "
    + oUser.Email + ", " + oUser.LoginName + Environment.NewLine);
}
// overloading methods to Check all type of files
public static void CheckUniquePermission(SPWeb webSP, SPWeb web){...}
public static void CheckUniquePermission(SPDocumentLibrary dcLibrary, SPWeb web
){...}
public static void CheckUniquePermission(SPFolder FolderName,SPFile file, SPWeb
web){...}
public static void CheckUniquePermission(SPList list, SPWeb web){...}

```

Výpis 9: Ukážka kódu filtrujúcich Administrátorov

5.2.5 Problémy pri fázach migrácie

Po dokončení a otestovaní skriptu som dostal za úlohu migrovať 10 fáz po zhruba 800 stránkach. Prvý problém, na ktorý som narazil, boli súbory, ktoré mali status 'check out'. Tieto súbory užívatelia mali otvorené a nesprávne ich uzavreli. Na výber boli tri riešenia:

Prvé riešenie bolo kontaktovať užívateľov a vyžiadať posledné zmeny na súboroch. Keďže posledná modifikácia na týchto stránkach bola pred viac ako tromi rokmi, užívatelia by zrejme ani nevedeli, že nejaký súbor mali otvorený.

Druhý spôsob bolo dostať stránku z readonly modu a uzavrieť súbor programovo. Tento spôsob, ale nebol schválený mojím nadriadeným, takže tiež nemohol byť povolený.

Finálne riešenie bolo zobrať staršiu verziu súboru a migrovať ju ako aktuálnu. Pokiaľ mal súbor len jednu verziu, tak cez migráciu neprešiel a bol považovaný za neexistujúci.

```

if (file.CheckOutStatus != SPFile.SPCheckOutStatus.None){
    // if file has only one version, ignore it
    if (file.Versions.Count <= 1){
        continue;
    }
    // get second last version of file
    fileFromSP = file.Versions[file.Versions.Count - 2].OpenBinaryStream();
}

```

Výpis 10: Ukážka kódu riešenia problému s nesprávne uloženými súbormi

Ďalší problém sa vyskytol pri súboroch, ktoré neobsahovali žiadne dáta. Ich veľkosť v bajtoch bola 0. C# nedokázal takýto file prečítať a zahlásiť chybu. Ošetrovanie podmienkou na veľkosť súboru vyriešilo tento problém. Pri migrácií som sa stretol aj s chybami na attachmentoch alebo listoch. Pri vstupe na danú stránku nebol žiaden z požadovaných view na liste alebo prílohe ku listu.

Zmapoval som si stránku na môj virtuálny stroj a skontroloval ju v prehliadači. Tam som dané chýbajúce súbory nenašiel, takže som migroval stránku bez chybných súborov. V neskoršej fázy som sa stretol s problémom dĺžky mena stránky. Maximálna dĺžka cesty k súboru je pre Windows 7 je 255 znakov. Po presiahnutí tejto dĺžky sa mi nepodarilo vytvoriť ACL a zip súbory. Rozhodol som sa meno každej stránky ohraničiť na dĺžku 45 znakov. Originálne meno stránky potom bude v ACL súbore a Site mape.

Po dokončení všetkých fáz budem ďalej pracovať na skripte, ktorý týmto stránkam odoberie práva užívateľov. Následne po troch mesiacoch Tieto stránky budú považované za neaktívne a ja dostanem za úlohu vytvoriť skript na ich vymazanie zo systému. Tým bude migrácia stránok plne dokončená a jedine čo zostane po pasívnych sieťach budú moje ACL a zip súbor s obsahom.

5.2.6 Doplnujúce informácie k projektu Migrácie Tieto stránok

- Výsledky zipovania boli vždy náhodne testované mojimi kolegami a vedúcou projektu. - Celý kolobeh tohto projektu bol opäť iteratívnym SCRUM systémom

6 Celkové zhodnotenie praxe

6.1 Uplatnenie teoretických a praktických znalostí nadobudnutých v škole

Na mojej praxi som zo začiatku využil objektovo orientované programovanie z predmetu programovacie jazyky 2, neskôr mi dosť pomohli vedomosti z predmetu AT.NET, čo sa týka rôznych konštrukcií jazyka c#, taktiež som využil nadobudnuté znalosti z predmetu algoritmy 2. Po príchode do firmy som dostal pracovný notebook, pripojený do siete. Pripájanie a vytváranie na virtuálne stroje nás učili na správe systému windows a bez týchto vedomostí by som nedokázal pracovať na žiadnom projekte. Znalosti z predmetu úvod do databátabázových systémov a databázové a informačné systémy mi v pokročilom štádiu mojej výuky v škole pomohli robiť dotazi na SP databáze. Predmet softwarové inžinierstvo mi pomohlo lepšie pochopiť scrum interačný proces vyvíjania software-u.

6.2 Chýbajúce znalosti a schopnosti

Medzi chýbajúce znalosti by som rozhodne pridal nedostatočnú znalosť asp.net, v škole síce bol na jednej hodine z predmetu predmete AT.NET úvod do web forms, každopádne na prácu na profesionálnej úrovni to nestačilo, väčšinou som čerpal vedomosti z samoštúdia popri škole. Taktiež by som ocenil v škole predmet zameraný na webové frameworky a všeobecný prehľad o nich.

6.3 Nadobudnuté znalosti

K nadobudnutým znalostiam by som určite chcel zaradiť základy platformy SP, pokročilú znalosť jazyka C# a taktiež pokročilú znalosť asp.net. Ďalej lepšie pochopenie SCRUM postupu vývoja software.

7 Záver

Na začiatku praxe som mal trochu problémy nadobudnúť potrebné znalosti pre prácu na zákazníckom projekte Katrin, ale postupom času a mentorovým usmernením sa mi podarilo úspešne začleniť do tímu. Zákaznícky projekt úspešne skončil v daný čas a to aj mojou zásluhou. Zákazník bol s mojou prácou spokojný, čo sa odzrkadlilo aj na mojom feedbacku. Z prvého projektu som si odniesol mnoho skúseností a plnohodnotné miesto v tíme.

Pri druhom projekte som sa stretol aj s seperátnymi skriptami a čiastočnou individuálnou prácou. Tu už moja práca neobsahovala žiadne časti mojích spolupracovníkov. Aj napriek tomu tu ale boli riešenia problémov na mítingoch a spoločný postup. Z druhého projektu si odnášam veľa skúsenosti ohľadom individuálnej práce a miesto zamestnanca v firme Tieto Czech s.r.o.

Celkovo hodnotím túto bakalársku prax veľmi pozitívne. Účelom bolo ma zaškoliť a pripraviť na budúcu prácu developera a myslím že tento cieľ som splnil. Firma Tieto mi ponúkla miesto developera druhej úrovne a momentálne pracujem na skrátený úväzok zamestnaneckej zmluvy.

Teším sa na ďalšiu spoluprácu s touto spoločnosťou a som vďačný za možnosť praxe, ktorú mi umožnila.

Literatúra

- [1] Tieto History [online] [cit. 2016-04-20]. Dostupné z: <http://www.Tieto.com/about-us>.
- [2] SharePoint history [online] [cit. 2016-04-20]. Dostupné z: <https://www.pilothouseconsulting.com/sharepoint-history-sharepoint-2016-release.html>.
- [3] Inside Microsoft SharePoint 2013
Hillier Scot, Olst Mirjam van, Pattison Ted, Andrew CONNELL, Wictor WILEN, Kyle DAVIS [cit. 2016-04-20].
- [4] Scrum software development [online] [cit. 2016-04-20]. Dostupné z: <http://scrummethodology.com>.
- [5] JIRA software pre vývoj produktov [online] [cit. 2016-04-20]. Dostupné z: <http://www.myjira.cz/novinky/predstaveni-jira-software.html>.
- [6] Team Foundation Server [online] [cit. 2016-04-20]. Dostupné z: <https://www.visualstudio.com/en-us/products/tfs-overview-vs.aspx>.